RECEIVED
MAY 2 1 2001
TECHNOLOGY CENTER R3700

GAME APPARATUS, SPECIFIED POSITION DETERMINING METHOD AND RECORDING MEDIUM AND PROGRAM

CROSS-REVERENCE TO RELATED APPLICATIONS

This application is based on upon and claims the benefit of priority from the prior Japanese Patent Application No. 2000-066878, filed May 10, 2000, the entire contents of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to a technique for executing a video game that displays an image of a predetermined map and an image of a cursor displayed on the image of the map and that carries out a game as specifying a predetermined position on the image of the map with the cursor.

Description of the Related Art

As such a game that carries out a game as specifying a predetermined position on the image of the map with the cursor, for example, there is a game, which is referred to as a war simulation game.

The game of this kind is designed to display a map, which is divided into numerous small regions formed in, e.g., a matrix, and images of combat elements placed in some small regions to combat with each other. Then, a combat is deployed as moving the combat elements appropriately in accordance with an operation content inputted by a game player. In other words, this kind of game is

designed to display a map screen page like a chessboard and combat elements like chess pieces on a display unit and repeats the combat between the combat elements, providing the game player enjoyment in a combat deployment and the strategy in response to the deployment.

Here, in such a game, the cursor for specifying the position of the combat element must be overlaid on the map image. Also, it is necessary to determine as to which position on the map is specified by the cursor.

By the way, in the conventional war simulation game, the map has only two-dimensional information in many cases. In a case where a two-dimensional map is used, it is easily figured out where the map and the cursor displayed two-dimensionally are positioned and determination as to which portion on the map is specified is easily carried out. Namely, the cursor displayed on the display unit always specifies the position on the map corresponding to the position of the map image displayed in a state that the cursor is superimposed thereon. Accordingly, in order to detect the position on the map, which the cursor specifies, it is enough that the position of the map image, which is superimposed on the image of the cursor on the image displayed on the display unit, may be detected. This makes it possible to detect which position on the map is specified by the cursor.

However, in a case where a three-dimensional map is used, the circumstances are not as simple as the use of the two-dimensional map.

Namely, in the case of the three-dimensional map, a display map, which has been converted to be two-dimensional, is displayed on the display unit. In this case, distortion, which results from the conversion, and a non-displayed portion occur depending on the case. Accordingly, unlike the case of two-dimensional map, it is impossible to detect as to which position on the map that the cursor specifies by only detecting the position where the display map image displayed on the display unit and the cursor image are superimposed on each other.

In a case where the map has three-dimensional information, the use of the map and the cursor both having three-dimensional information makes it possible to obtain the cursor position on the map by computation.

However, in the case where such a technique is adopted, there occurs a necessity to conform the positional relationship between upper and lower and right and left on the map image displayed on the display unit to the positional relationship between upper and lower and right and left on the moving direction of the cursor. This generates influence, which is unfavorable to the game player, in some cases. Namely, there is a case in which the aforementioned map image is displayed on the display unit in the direction that the positional relationship between upper and lower and right and left on the map image displayed on the display unit does not always instinctively conform to the positional relationship between upper and lower and right and left on the moving direction of the cursor with the intention of improving flexibility of game deployment or

making the image visible to the game player. In such a case, the game player cannot instinctively perform the input in the moving direction of the cursor.

SUMMARY OF THE INVENTION

It is an object of the present invention to provide a technique for solving the aforementioned problem. More specifically, an object of the present invention is to provide a technique for determining as to which position on a 3D map is specified by a cursor in a game apparatus that executes a game by displaying a display map image based on the 3D map, including three-dimensional information, and a cursor, including two-dimensional information and indicating a predetermined position of the 3D map, on a display unit.

According to a first aspect of the present invention, there is provided a specified position determining method applied to a game apparatus, comprising the steps of: generating map data to display a map image on a display of the game apparatus, the map image two-dimensionally expressing a corresponding three-dimensional map which includes information representing a predetermined three-dimensional field; generating cursor data to display a cursor on the displayed map image; controlling a position of the displayed cursor in accordance with an instruction from an operator; virtually disposing the three-dimensional map in parallel to the map image at a backward position thereof seeing from a predetermined viewpoint, such that straight lines extending from the predetermined viewpoint to given points on a peripheral edge of the map image further pass through corresponding points on a peripheral edge of the three-

dimensional map; projecting the predetermined viewpoint onto the three-dimensional map via a position of the cursor displayed on the map image; and detecting a point on the three-dimensional map where the projected viewpoint intersects the predetermined three-dimensional field, whereby determining the detected point as a position where the cursor specifies on the displayed map image.

It may be arranged that the map data generating step includes a substep of generating map data to display a position on the map image, which corresponds to the determined position, on the display to be distinguishable from other positions.

It may be arranged that the predetermined three-dimensional field includes a plurality of areas, and the detecting step includes a substep of detecting which of the plurality of areas includes the detected point. In this case, the map data generating step may include a substep of generating map data to display an area on the map image, which corresponds to the detected area, on the display to be distinguishable from other areas.

Therefore, not only the game player specifies which position on the 3D map but also the determination result can be displayed on the display unit to be visually recognized by the game player. This makes it possible for the game player to easily grasp which position is specified by the cursor and to make the game more easily understandable.

It may be arranged that the predetermined three-dimensional filed represents at least one of a ground surface and a water surface.

According to a second aspect of the present invention, there is

provided a game apparatus comprising: a generator for generating map data to display a map image on a display of the game apparatus, the map image two-dimensionally expressing a corresponding three-dimensional map which includes information representing a predetermined three-dimensional field, generating cursor data to display a cursor on the displayed map image, and controlling a position of the displayed cursor in accordance with an instruction from an operator; and a controller for executing game processing in accordance with a position on the displayed map image specified by the cursor, wherein the generator virtually disposes the three-dimensional map in parallel to the map image at a backward position thereof seeing from a predetermined viewpoint, such that straight lines extending from the predetermined viewpoint to given points on a peripheral edge of the map image further pass through corresponding points on a peripheral edge of the three-dimensional map, projects the predetermined viewpoint onto the three-dimensional map via a position of the cursor displayed on the map image, and detects a point on the three-dimensional map where the projected viewpoint intersects the predetermined three-dimensional field, whereby determining the detected point as a position where the cursor specifies on the displayed map image.

Accordingly, even when the 3D map (three-dimensional map), which is useful to improve enjoyment in the game, is used, the position on the 3D map can be specified by the two-dimensional cursor that moves independent of the 3D map. This eliminates the need for conforming the upper and lower and right and left

directions on the 3D map to the upper and lower and right and left directions in the moving direction of the cursor. Therefore, the game player can instinctively specify the position using the cursor.

The 3D map may include three-dimensional information of terrain roughness.

Furthermore, the cursor position determining method and the game apparatus according to the present invention can be implemented by reading a program code recorded on a computer-readable storage medium set forth below into a predetermined computer, for example, an exclusive computer game and a home computer.

According to a third aspect of the present invention, there is provided a storage medium having computer readable program code means embodied in the medium, the computer readable program code means comprising: computer readable program code means for generating map data to display a map image on a display of the game apparatus, the map image two-dimensionally expressing a corresponding three-dimensional map which includes information representing a predetermined three-dimensional field; computer readable program code means for generating cursor data to display a cursor on the displayed map image; computer readable program code means for controlling a position of the displayed cursor in accordance with an instruction from an operator; computer readable program code means for virtually disposing the three-dimensional map in parallel to the map image at a backward position thereof seeing from a predetermined viewpoint, such that

straight lines extending from the predetermined viewpoint to given points on a peripheral edge of the map image further pass through corresponding points on a peripheral edge of the three-dimensional map; computer readable program code means for projecting the predetermined viewpoint onto the three-dimensional map via a position of the cursor displayed on the map image; and computer readable program code means for detecting a point on the three-dimensional map where the projected viewpoint intersects the predetermined three-dimensional field, whereby determining the detected point as a position where the cursor specifies on the displayed map image.

According to a fourth aspect of the present invention, there is provided a computer program for a computer having a display, the computer program causing the computer to execute the steps of: generating map data to display a map image on a display of the game apparatus, the map image two-dimensionally expressing a corresponding three-dimensional map which includes information representing a predetermined three-dimensional field; generating cursor data to display a cursor on the displayed map image; controlling a position of the displayed cursor in accordance with an instruction from an operator; virtually disposing the three-dimensional map in parallel to the map image at a backward position thereof seeing from a predetermined viewpoint, such that straight lines extending from the predetermined viewpoint to given points on a peripheral edge of the map image further pass through corresponding points on a peripheral edge of the three-dimensional

map; projecting the predetermined viewpoint onto the three-dimensional map via a position of the cursor displayed on the map image; and detecting a point on the three-dimensional map where the projected viewpoint intersects the predetermined three-dimensional field, whereby determining the detected point as a position where the cursor specifies on the displayed map image.

It may be arranged that the computer program is stored in a computer readable storage medium.

It is noted that the program code described in this specification is a concept including data necessary for causing the game apparatus to execute the game, a control parameter, and the like in addition to the program itself.

BRIEF DESCRIPTION OF THE DRAWINGS

These objects and other objects and advantages of the present invention will become more apparent upon reading of the following detailed description and the accompanying drawings in which:

FIG. 1 is a hardware configuration view illustrating an example of a game apparatus main body to which the present invention is applied;

FIG. 2 is a function block diagram illustrating a configuration example of a game apparatus according to a first embodiment of the present invention;

FIG. 3 is a function block diagram illustrating a configuration of a game controller shown in FIG. 2;

FIG. 4 is a function block diagram illustrating a configuration of an image generator shown in FIG. 2;

FIG. 5 is a flowchart of a cursor position determining method of this embodiment;

FIG. 6 is a view illustrating one example of an image displayed on a display unit in a normal state;

FIG. 7 is a view conceptually illustrating an outline of a cursor position determining method of this embodiment; and

FIG. 8 is a view illustrating one example of an image displayed on a display unit in a combat state;

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

A preferred embodiment of the present invention will be specifically described with reference to the accompanying drawings.

A game apparatus of the present invention will be first explained. The game apparatus of the present invention is implemented as a game apparatus having hardware configured as illustrated in FIG. 1.

A game apparatus main body 2 has two buses of a main bus B1 and a sub-bus B2 as illustrated in the figure. These buses B1 and B2 are connected or disconnected to/from each other via a bus interface INT.

A main CPU (Central Processing Unit) 10, which is composed of a microprocessor, a first vector processing unit (VPU0, hereinafter referred to as "first VPU") 20, a main memory 11 having a RAM (Random Access Memory), a main DMAC (Direct Memory Access Controller) 12, an MPEG (Moving Picture Experts Group) decoder (MDEC) 13, a second vector processing unit (VPU1, hereinafter

referred to as "second VPU") 12, and a GIF (Graphical Synthesizer-interface) 30, which functions as an arbiter for the first and second VPUs 20 and 21, are connected to the main bus B1. Moreover, a graphical synthesizer (hereinafter referred to as "GS") 31 is connected thereto via the GIF 30. A CRTC (CRT controller) 33 for generating a video output signal is connected to the GS31.

Though the CRTC is connected to a predetermined display unit for displaying a game image, the illustration of this display unit is omitted.

The main CPU 10 reads in an activation program from a ROM 17 on the sub-bus B2 via the bus interface INT at a game apparatus start-up time, and executes the activation program to operate an operating system. The main CPU 10 also controls a media drive 60 to read out an application program and data from a media 61 to store them to the main memory 11. Moreover, the main CPU 10 performs geometry processing with respect to various kinds of data read from the media 61, for example, three-dimensional object data (coordinate values of vertices (representative points) of polygons) composed of a plurality of basic graphics (polygons) in cooperation with the first VPU 20.

Additionally, in the main CPU 10, there is provided a high-speed memory, which is referred to as SPR (Scratch Pad RAM), for temporarily storing the result of processing in cooperation with the first VPU 20.

The first VPC 20 has a plurality of operators to carry out an operation of a real number of floating-point arithmetic. The first

VPC 20 performs floating-point arithmetic in parallel using these operators. Namely, the main CPU 10 and first VPC 20 perform arithmetic processing, which requires a precise operation on a polygon basis, among geometry processing. Then, they generate a display list having polygon definition information as its content such as a vertex coordinate sequence, shading mode information and so on obtained by this processing.

Polygon definition information is composed of drawing region setting information and polygon information. Drawing region setting information is made up of an offset coordinate at a frame buffer address of the drawing region and a coordinate of a drawing clipping region to cancel the drawing when a polygon coordinate is placed at the outside of the drawing region. Polygon information is composed of polygon attribute information and vertex information. Polygon attribute information is information for specifying a shading mode, an α blending mode, and a texture mapping mode. Vertex information includes a vertex drawing inner region coordinate, a vertex texture inner region coordinate and a vertex color, etc.

As is the case with the first VPU 20, the second VPU 21 has a plurality of operators to carry out an operation of a real number of floating-point arithmetic. The second VPC 21 performs floating-point arithmetic in parallel using these operators. Then, the second VPC 21 generates a display list having an image generated by an operation of a control unit 81 and an operation of a matrix, e.g., relatively easy two-dimensional polygon definition information as its content, which can be generated by processing of, e.g., perspective

conversion with respect to a simple-shaped object such as a building, a car, etc., a parallel light source calculation, generation of a two-dimensional curvature, and so on.

Though the first VPU 20 and second VPU 21 have the same configuration, they function as a geometry engine that starts arithmetic processing with a different content. Normally, processing (non-fixed geometry processing) for character movement, which needs a complicated behavior calculation, is assigned to the first VPU 20. Then, processing (fixed geometry processing) for an object, which is simple but needs numerous polygons, for example, a background building and the like, is assigned to the second VPU 21.

The first VPU 20 performs macro arithmetic processing to be synchronized with a video rate, and the second VPU 21 is designed to operate in synchronization with the GS 31. For this purpose, the second VPU 21 is equipped with a direct path directly connecting to the GS31. Conversely, the first VPU 20 is closely connected to a microprocessor provided in the main CPU 10 to facilitate programming of complicated processing.

The first VPU 20 and second VPU 21 generate the display lists. The generated display lists are transferred to the GS 31 via the GIF 30.

The GIF 30 performs arbitration to prevent occurrence of a collision at the time of transferring the display lists generated by the first VPU 20 and second VPU 21 to the GS 31. In this embodiment, however, a function of checking these display lists in order of priority to transfer them to the GS31 from the higher display list is

added to the GIF 30. Information indicating priority of the display lists is generally described in its tag region when the VPUs 20 and 21 generate display lists, but the GIF may determine the priority originally.

The GS 31 holds drawing contexts. The GS 31 reads the corresponding drawing context based on identification information of image contexts included in the display lists notified from the GIF 30. Then, the GS 31 performs rendering processing by use of the read drawing context to draw a polygon in a frame memory 32. The frame memory 32 can be used as a texture memory, so that a pixel image on the frame memory is also used as a texture to be adhered onto a polygon to be drawn.

Referring back to FIG. 1, the main DMAC 12 provides DMA transfer control to the respective circuits connected to the main bus B1, while provides DMA transfer control to the respective circuits connected to the sub-bus B2 in accordance with the state of bus interface INT.

The MDEC 13 operates concurrently with the main CPU 10, and decompresses data compressed in MPEG (Moving Picture Experts) or JPEG (Joint Photographic Experts Group).

A sub-CPU 14 comprising a microprocessor and so on, a sub-memory 15 having a RAM, a sub-DMAC 16, a ROM 17 having a program such as an operating system and the like stored therein, a sound processing unit (SPU) 40 that reads sound data stored in a sound memory 59 to output as an audio output, a communication controller (ATM) 50 that performs data transmission/reception via a

public switched telephone network, a media drive 60 for attaching a recording medium 61 such as a CD-ROM and DVD-ROM, and an input unit 70 are connected to the sub-bus B2. The input unit 70 includes a connection terminal 71 for connecting a control unit 81, a connection terminal 72 for connecting a memory card MC, a video input circuit 73 for inputting image data from an external unit, and an audio input circuit for inputting sound data from an external unit.

The control unit 81 corresponds to inputting means of the present invention.

The sound processing unit 40 is connected to a speaker (not shown) of the present invention to output effective sounds.

The sub-CPU 14 performs various kinds of operations in accordance with a program stored in the ROM. The sub-DMAC 16 provides the DMA transfer control to the respective circuits connected to the sub-bus B2 only when the main bas B1 and sub-bus B2 are separated from each other by the bus interface INT.

The game apparatus of this embodiment also performs characteristic geometry processing. As mentioned above, the concurrent operation of first VPU 20 and second VPU 21 makes it possible to perform geometry processing, which is adaptable to high-speed rendering processing. Hereinafter, as to whether the result of arithmetic processing obtained by the first VPU 20 is directly sent to the GIF 30 or serially sent thereto via the second VPU 21, the selection is designed to be performed by software. The former mode is referred to as a parallel connection and the latter

mode is referred to as a serial connection. In either case, the output of second VPU 21 has a path directly connected to the GIF 30 (GS31) and performs a coordinate conversion in synchronization with timing of rendering processing of the GS 31. As a result, the standby state of GS 31 is not maintained more than necessary.

In the above-arranged game apparatus, when the recording medium 61 of the present invention having, for example, a CD-ROM is attached to the disk drive 41 and power is turned on or reset processing is performed, the main CPU 10 executes OS recorded in the ROM 17. When OS is executed, the main CPU 10 initializes the entirety of the apparatus to confirm the operation and the like and controls the disk controller 40 to execute a game program recorded in a CD-ROM 44 after reading it to the main memory 13. The execution of this game program allows the main CPU 10 to form the function block shown in FIG. 2 and to implement the game apparatus 2 of the present invention.

As illustrated in FIG. 2, the game apparatus of this embodiment comprises an instruction information analyzer 110, a game controller 120, and an image data generator 130.

The instruction information analyzer 110 is connected to the control unit 81, and determines the operation content inputted from the control unit 81 by the game player. The operation content analyzed by the instruction information analyzer 110 is sent to the game controller 120 to be reflected in control of executing the game.

The game controller 120 integrally controls the entirety of game to be executed by the present invention, and corresponds to control

means of the present invention.

In addition, the game executed by the game apparatus of the present invention is a war simulation game. The war simulation game to be executed by this game apparatus can be described as follows:

A map image of a map, which is divided into rectangular small regions formed in a matrix, is displayed on the display unit and combat elements, which fight with each other, are displayed on some small regions so that the combat is deployed as moving the combat elements appropriately in accordance with instructions from the game player. This war simulation game is executed by alternately repeating a normal state and a combat state wherein the normal state indicates that movement of the combat elements is performed and the combat state indicates that the combat elements fight with each other in the normal state.

The game controller 120 comprises a selector 121, a normal state controller 122, and a combat state controller 123 as shown in FIG. 3.

The selector 121 selects the normal state or combat state in accordance with the operation content inputted by the game player via the control unit 81.

The normal state controller 122 performs control in the normal state. The combat state controller 123 performs control in the combat state. Data of each of the normal state controller 122 and the combat state controller 123 is sent to an image data generator 130 to be described later to be reflected in generation of an image to

be displayed on the display unit.

The combat state controller 123 comprises a distance detector 123A, which corresponds to distance detecting means of the present invention. The distance detector 123A is capable of receiving data selected from the selector 121. At the time of receiving the selected data, the distance detector 123A detects a distance between the player's combat element to execute the combat according to the player's operation content inputted by the game player among the player's combat elements and the opponent's combat element fighting with the player's combat element, whereby generating distance data of the corresponding distance. The generated distance data is sent to the image generator 130.

The image data generator 130 generates image data to display a desired image on the display unit. The display unit displays an image based on this image data. This image data generator 130 corresponds to image data generating means of the present invention.

The image data generator 130 comprises a normal state image determining unit 131, a combat state image determining unit 132, and a data generator 133 as shown in FIG. 4. The normal state image determining unit 131 determines an image to be displayed on the display unit in the normal state. The combat state image determining unit 132 determines an image to be displayed on the display unit in the combat state. The data generator 133 generates final image data based on data, which is indicative of the image to be displayed on the display unit and which is sent from either the

normal state image determining section 131 or the combat state image determining section 132.

The normal state image determining section 131 includes a map data generator 131A, a cursor data generator 131B, a specified position determining unit 131C and a 3D map data recorder 131D.

The 3D map data recorder 131D records 3D map data.

This 3D map data is data of a map, which includes three-dimensional information and which is used to generate map data about a display map to be display on the display unit in the normal state. The 3D map data is designed to generate such a display map that is divided into small quadrangle regions formed in a matrix though data is not limited to this.

The map data generating section 131A generates map data about a two-dimensional display map to be displayed on the display unit based on 3D map data read from the 3D map data recording section 131D. The map data is sent to the data generator 133.

The map data generator 131 also generates combat element data for displaying the player's combat element and the opponent's combat element on the display wherein the player's combat element moves based on the operation content inputted by the gamer player via the control unit 81 and the opponent's combat element moves automatically upon receipt of control of predetermined data. It is noted that the opponent's combat element may execute movement and combat automatically under control of the game controller 120 or based on the operation content inputted by the other player.

The combat element described herein indicates one that is

movable on the 3D map and fights with an enemy combat element. For example, the combat element is expressed as a tank, a flat land, a foot soldier, etc, depending on the kinds of war simulation games.

The cursor data generator 131B displays a two-dimensional cursor at a predetermined position on the screen based on the operation content inputted by the player via the control unit 81. The cursor data generator 131B generates cursor data for displaying the cursor on the screen. The data is sent to the data generator 133.

The specified position determining unit 131C determines a position on the 3D map specified by the cursor. Position data about the position specified by the cursor is sent to the data generator 133.

The combat state image determining unit 132 determines an image to be displayed on the display in the combat state. Specifically, the combat state image determining unit 132 determines what image should be displayed on the display based on data generated by the combat state controller 123.

Actions of the game apparatus of this embodiment will be next explained in connection with one embodiment of the specified position determining method according to the present invention. This game apparatus executes the specified position determining method of the present invention with a flow shown by FIG. 5.

Power source is turned on or reset processing is performed, and the game player inputs the operation content of the game start via the control unit 81, whereby the game apparatus starts a war simulation game (S201).

When the game is started, the normal state is initiated (S202),

so that the screen page of the normal state is displayed on the display unit. Namely, the selector 121 selects the normal state in an initial state.

On the screen page of the normal state, the display map, which expresses the 3D map two-dimensionally, the player's combat elements, the opponent's combat elements, and the cursor are displayed. An example of the image displayed on the display unit at this time is illustrated in FIG. 6.

The display map, combat elements, and cursor are displayed as follows:

Specifically, the map data generator 131A reads 3D data map including three-dimensional data including three-dimensional data from the 3D map data recorder 131D, and converts it to map data about two-dimensional display map, whereby generating the display map.

A display map M is as shown in FIG. 6. Namely, the display map M is divided into quadrangle small regions M1, M1, ... in a matrix form. In this example, different geographical features such as a flat ground, a river, a mountainous district, and the like are given to the respective small regions M1, M1, ... These features exert an influence upon the result of combat at each of the small regions M1, M1, ... That is, the combat elements S1, S1, ... have intrinsic properties, respectively, and advantageous small regions M1, M1, ... and disadvantageous regions M1, M1, ... are predetermined. The game player works out a strategy of movement of each of player's combat elements S1, S1, ... with consideration given to such

geographical features.

These combat elements S1, S1, ... are designed to move in accordance with data generated by the instruction information analyzer 110 based on the operation content inputted by the player via the control unit 81. The map data generator 131A generates combat element data for displaying the respective combat elements S1, S1, ... on the display unit based on data generated by the instruction information analyzer 110 under the normal state controller 122.

Sub. a1> ~~It is noted that the respective combat elements S1, S1, ... are designed to move in the unit of small regions M1, M1, ... as a minimum unit. The opponent's combat elements M1, M1 ... move in the unit of small regions M1, M1... under the normal state controller 122, and the player's combat elements are designed to move to the small regions M1, M1, ..., which the player has specified with the cursor C.~~

The cursor C is displayed based on cursor data generated by the cursor data generator 131B. The cursor data originally has only two-dimensional information and is expressed two-dimensionally.

The cursor data is generated by the cursor data generator 131B based on the operation content inputted by the game player via the control unit 81. In accordance with the instructions of movement in upper and lower and right and left directions inputted by the game player using the controller, the cursor C is controlled to be moved to the corresponding direction. Data necessary for this control is inputted to the cursor data generator 131B via the control unit 81,

the instruction information analyzer 110, and the normal state controller 122.

Axes in the upper and lower and right and left directions where the cursor C moves are directions indicated by XC and XY in the figure, and they deviate from axes XM and YM in the upper and lower and right and left directions about the display map M. Namely, the moving direction of the cursor C is controlled regardless of the axes XM and YM in the upper and lower and right and left directions on the display map M.

In the normal state, it is also determined as to which position on the 3D map the cursor C specifies using the display map M. Namely, when the game player moves the player's combat elements S1, S1, ..., small regions M1, M1, .., must be specified by the cursor C. Accordingly, the specified position determining unit 131C determines the position on the 3D map, which the cursor C specifies on the display map M, by the way set forth below.

The specified position determining unit 131C determines not only the position on the 3D map specified by the cursor C using the display map M but also which small region M1 the cursor C specifies using the display map M.

This determination is performed by the method as shown in FIG. 7. It is noted that FIG. 7 schematically and conceptually shows this determination method.

According to this method, an image G, which is displayed on the display unit, and a predetermined viewpoint E, which is placed in front of the image G, are virtually formed. Next, a rectangular 3D

map 3DM is virtually (imaginarily) formed such that four corners thereof are located on straight lines originating from the viewpoint E and passing through four corners of the image G. Accordingly, as seen from FIG. 7, the 3D map 3DM is placed in parallel with the image G at a backward position seeing from the viewpoint E. More specifically, the 3D map 3DM is virtually disposed such that an X-Y plane (a plane including X and Y axes) thereof is set in parallel to the image G which is two-dimensional. This 3D map corresponds to the display map M displayed on the display unit. At this time, the specified position determining unit 131C uses, for example, 3D map data obtained from the 3D map recorder 131D via the map data generator 131A in order to form the 3D map 3D virtually.

Then, in this state, a straight line passing through the viewpoint E and a predetermined portion of the cursor C (e.g., top portion of the cursor C indicated by an arrow shape) is virtually formed, i.e. the viewpoint E is projected via the predetermined portion of the cursor C. Next, it is detected which position on the 3D map 3DM the straight line intersects. After that, it is also detected in which small region M1 a point of such an intersection exists.

Then, by assuming that the cursor C specifies the small region M1 where the point of intersection exists, it is determined which small region M1 the cursor C specifies using the display map M.

In this embodiment, data of which small region M1 the cursor C specifies is sent to the map data generator 131A. As a result, the map data generator 131A uses the above data in generating combat

element data after deciding destinations of combat elements S1, S1.

The above data can be used as follows though this is not always needed.

Sub a2> Specifically, the above data can be used to generate image data for displaying a small region S1 specified by the cursor C at the current point on the display unit as distinguished from other small regions S1, S1, ... This can be attained by, for example, making a difference in color between the small region S1 and other regions S1, S1... or adding some pattern to the small region S1 though the illustration is omitted. Data generated in this case is capable of displaying such a color and a pattern.

?

The image displayed on the display unit in the normal state is displayed based on image data generated when the data generator 133 combines map data, combat element data and cursor data generated as mentioned above.

In the normal state, a determination whether or not a predetermined condition about the start of combat is satisfied is performed (S203). For example, this condition is satisfied when the game player inputs the operation content, which corresponds to the instructions in which the predetermined player's combat elements S1, S1 are fought with the opponent's combat elements S1, S1, via the control unit 81 or when the normal state controller 122 supports the start of combat in accordance with the predetermined condition. The determination whether or not this condition is satisfied is performed by, for example, the selector 121.

When this condition is satisfied (S203: YES), the selector 121

generates switching data for switching from the normal state to the combat state and the combat state controller 121 receives it and starts to control the combat state, so that the combat state is started (S204). When the condition is not satisfied (S203: NO), the normal state is continued.

Sub. a3> When the combat state is started, an image, which shows a situation in which the player's combat element S1 and the opponent's combat element S2 are fighting with each other, is displayed on the display unit in real time.

This image is displayed on the display unit in accordance with image data, which the data generator 133 has generated based on data determined by the combat state image determining unit 132 under control of the combat state controller 123. This image shows that the player's combat element executes an action in real time in accordance with the operation content inputted by the game player. Accordingly, the data generator 133 generates combat image data to display the player's combat element S1, which executes a combat in accordance with the operation content inputted by the game player among the player's combat elements S1, S1, and the opponent's combat element S1 fighting with the player's combat element S1 in real time. In other words, the player's combat element S1 is displayed on the screen in real time in accordance with the operation content inputted by the game player. For example, the image as a moving image using a polygon is displayed on the display unit. One example of the image displayed on the display unit in the combat state is illustrated in FIG. 8.

At the combat state starting time, the following processing is carried out, whereby determining an initial screen page at the beginning of the combat state.

First, when the condition for starting the combat state is met, the aforementioned combat data is inputted to the combat state controller 123. This data is also sent to the distance detector 123A provided in the combat state controller 123.

Upon receipt of this data, the distance detector 123A detects a distance between the player's combat element S1, which executes the combat, and the opponent's combat element S1 so as to generate distance data of the distance. This distance denotes, for example, a distance between both combat elements S1 on the 3D map 3DM.

Distance data may be one that is proportional to the distance between both combat elements S1. In this example, however, to simplify the explanation, two kinds of different distance data are generated depending on whether both combat elements S1, which are to fight with each other, are placed at the small regions M1 adjacent to each other via the side or at a different position. Namely, the distance detector 123A generates distance data indicating that both combat elements S1 are placed in a close distance range in the former case and in a long distance range in the latter case.

This can be explained with reference to FIG. 9. In this example, it is assumed that the opponent's combat element S1 is placed at the small region M1 as illustrated in ~~the~~ a figure. Distance data indicating that both are in the close distance range is generated

only when the player's combat elements are placed at four small regions shown by diagonal lines. When the player's combat elements are placed at the other regions, distance data indicating that both are in the long distance range is generated.

Then, this data is sent to the combat state image determining unit 132 provided in the image data generator 130. The combat state image determining unit 132 generates image data about the initial image displayed on the display unit in accordance with the distance data at the beginning of the change to the combat state. In this embodiment, when the combat state image determining unit 132 receives distance data indicating that both combat elements S1 are placed in the close distance range, the combat state image determining unit 132 is configured to generate initial image data for displaying the initial image indicating that both combat elements are in the close distance range as compared with the case in which it receives data indicating that both combat elements are in the long distance range.

More specifically, this is illustrated as in FIG. 10. FIG. 10 shows an example of the initial image in a case (A) in which the combat state image determining unit 132 receives distance data indicating that both combat elements S1 are placed in the close distance range and that of the initial image in a case (B) the combat state image determining unit 132 receives data indicating that both combat elements are in the long distance range. In either example of FIG. 10, the soldier displayed at the front indicates the player's combat element S1 and the solider displayed at the backward is the

opponent's combat element S1.

According to these examples, in the case (A) in which distance data indicates that both combat elements S1 are placed in the close distance range, the opponent's combat element S1 is displayed largely as compared with the case (B) in which distance data indicates that both combat elements S1 are placed in the long distance range.

In a case where distance data is data that shows numerous levels (or non-level), which are proportional to the distance, the distance between the player's combat element S1 and the opponent's combat element S1 can be variously expressed.

In the combat state, it is judged whether or not a fixed condition about the end of combat is met (S205). For example, when either of the player's combat element S1 and the opponent's combat element S1, which are fighting with each other, is completely destroyed or annihilated, this condition can be met. The judgment whether this condition is met or not is performed by, for example, the selector 121.

When this condition is met (S205: YES), the combat state is ended. When the combat state is ended, the selector 121 performs switching from the combat state to the normal state so as to start the normal state (S206). When this condition is not met (S205: NO), the combat state is continued.

In the normal state, it is judged whether or not a fixed condition about the end of the game is met (S207). For example, when all player's combat elements S1 or opponent's combat elements S1 are

completely destroyed or either the player or the computer acknowledge one's defeat, this condition can be met.

When this condition is met (S207: YES), the war simulation game is ended (S208). When the combat state is ended, an image indicating whether the player is a victor or a loser or an end roll is displayed on the display unit.

When this condition is not met (S207: NO), the war simulation game is continued.

As is obvious from the above explanation, according to the present invention, not only a 3D map having three-dimensional information and a cursor having only two-dimensional information can be displayed on the same screen page and but also their positions can be figured out, whereby determining as to which position on the 3D map is specified by the cursor.

This makes it possible to display the three-dimensional map and the two-dimensional cursor, which is in a state that it is completely separated from the map, on the display unit. As a result, enjoyment in the game can be increased without putting the load on the game player to understand the game.

Various embodiments and changes may be made thereunto without departing from the broad spirit and scope of the invention. The above-described embodiment intended to illustrate the present invention, not to limit the scope of the present invention. The scope of the present invention is shown by the attached claims rather than the embodiment. Various modifications made within the meaning of an equivalent of the claims of the invention and within

the claims are to be regarded to be in the scope of the present invention.